الجمهـــوريـــــــة الجزائـريـــــــة الديمـــقـــراطيـــــــة الشعبيـــــــة
People`s Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research
National School of Cyber Security
Foundation Training Department

وزارة التعـليـم العـــالـي والـبـحـــث الـعـلـمـي
المدرســـة الوطنيـــة العليـــا في الأمـن السيبـراني
قســـم التكويـــن القـــاعدي

Surname : ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ **First Name :** ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░

N° Student Card : ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ ░ **Date : 27 / 11 / 2025**

Section : ░ ░ ░ ░ **Group :** ░ ░

Module : Architecture of Computer 1 (Midterm) ░ ░ ░ ░ ░ ░ ░ **Academic Year : 2025 / 2026**

CODE : Reserved for administration

MARK :

/ 20

CODE:

reserved for administration

Marker's Fullname :

…………………………

…………………………

…………………………

Invigilator :

…………………………

…………………………

…………………………

…………………………

…………………………

…………………………

Exercise 1: (04 points)

Select all correct answers (multiple correct answers possible). +0.5 point for each correct answer.

1. The Simplified expression of $(A + B)(\bar{A} + B)(A + \bar{A})$ is:

a) A | c) B
b) A+B | d) 0

2. The symmetric function of 3 variables is 1 only if the number of 1s is even. The minimal SOP form is:

a) AB + AC + BC | c) A + B + C
b) $A \oplus B \oplus C$ | d) ABC

"None is correcte"

3. In IEEE 754 single precision, the hexadecimal number C1200000 **corresponds to :**

a) -10.0 | c) -9.5
b) -9.0 | d) -8.5

4. The biased exponent for the value 1.0 in IEEE 754 single precision is:

a) 127 | c) 0
b) 128 | d) 1

5. In The sum of two 8-bit binary numbers 01101101 + 11011011 is:

a) 01001000 (overflow) | c) 10100110 (overflow)
b) 01001000 (no overflow) | d) 10100110 (no overflow)

6. Which single logic gate can implement any Boolean function**?**

a) AND | c) NAND
b) XOR | d) OR

7. Given the Boolean expression F=A+AB, what is the dual **of this expression?**

a) $F_d = A\cdot(A+B)$ | c) $F_d = A+(A\cdot B)$
b) $F_d = A\cdot(A\cdot B)$ | d) $F_d = A\cdot(A+B)+B$

8. Which of the following statements about Karnaugh maps (K-maps) is TRUE**?**

a) K-maps can only be used for functions with 2 or 3 variables.
b) Adjacent cells in a K-map differ by exactly **one variable**.
c) K-maps can directly implement XOR functions without grouping.
d) The number of rows in a K-map is always equal to the number of variables.

This area will be cut.

Exercise 2: (2 points)

1. How many positive integers can be represented in binary using one byte?

The total number of positive integers that can be represented in binary using one byte is 2^8-1 = 255 (1pts)

……

2. How many bits are necessary to represent in binary the natural numbers less than or equal to n?

Let k be the number of bits needed to represent a natural number ≤ n.
We have $2^k - 1 \geq n \Rightarrow 2^k \geq n+1 \Rightarrow k \geq \log_2(n+1) \Rightarrow k=\lceil \log_2(n+1) \rceil$ (1pts)

Exercise 3: (03 points)

Consider the following two numbers encoded according to the IEEE 754 standard (32 bits) and represented in hexadecimal: 3EE00000 and 3D800000.

1. Compute their sum and give the result in IEEE 754 format and in decimal form.

3EE00000=00111110111000000000000000000000 = $(+1,110*2^{-2})_2$

3D800000=00111101100000000000000000000000 = $(+1, 0*2^{-4})_2$

3EE00000 + 3D800000 = $(+1,110*2^{-2})_2 + (+1,0*2^{-4})_2 = (+111,0*2^{-4})_2 + (+1, 0*2^{-4})_2 = (+1000,0*2^{-4})_2 = (+1,0*2^{-1})_2$

$(+1,0*2^{-1})_2$ = 00111111000000000000000000000000 = **3F000000** (1pts)

$(+1,0*2^{-1})_2 = (0.1)_2 = \mathbf{(0.5)_{10}}$ (0.5pts)

2. Same question for the numbers: C8800000 and C8000000.

C8800000= 11001000100000000000000000000000 = $(-1.0*2^{+18})_2 = (-10.0*2^{+17})_2$

C8000000= 11001000000000000000000000000000 = $(-1.0*2^{+17})_2$

C8800000 + C8000000 = $(-10.0*2^{+17})_2 + (-1.0*2^{+17})_2 = (-11.0*2^{+17})_2 = (-1,1*2^{+18})_2$

$(-1,1*2^{+18})_2$ = 11001000110000000000000000000000 = **C8C00000**(1pts)

$(-1,1*2^{+18})_2 = (-1,5*2^{+18})_{10} = \mathbf{(-393216)_{10}}$(0.5pts)

Exercise 4: (06 points)

In this exercise, numbers are stored in 1-byte words, i.e., 8 bits.

1. Give the two's complement representation of the following signed integers: -13 and -127.

$(-13)_{10}$ = $(10001101)_{SM}$ = $(11110010)_{1'c}$ = $(1111\ 0011)_{2'c}$ (0.5pts)

....

$(-127)_{10}$ = $(11111111)_{SM}$ = $(10000000)_{1'c}$ = $(10000001)_{2'c}$ (0.5pts)

2. Compute the opposite of the following signed integers represented in 1 byte: 10011101 and 00110011.

$(10011101)_{2's}$ = $(10011100)_{1's}$ = $(11100011)_{SM}$ = $(-99)_{10}$ its opposite is $(+99)_{10}$ = $(01100011)_{SM}$ (0.5pts)

....

$(00110011)_{SM}$ = $(+51)_{10}$ its opposite is $(-51)_{10}$ = $(10110011)_{SM}$ = $(11001100)_{1'c}$ = $(11001101)_{SM}$ (0.5pts)

....

3. Give the decimal representation of the following signed integers (in two's complement): 11001101 and 00001101.

$(11001101)_{2's}$ = $(11001100)_{1's}$ = $(10110011)_{SM}$ = $(-51)_{10}$ (0.5pts)

....

$(00001101)_{2's}$ = $(00001101)_{SM}$ = $(+13)_{10}$ (0.5pts)

....

4. Convert to binary, then compute on 8 bits the following operation: (-13) + 13, 23 - 46, and 127 + 2.

$(+13)_{10}$ = $(00001101)_{SM}$, $(-13)_{10}$ = $(10001101)_{SM}$ = $(11110010)_{1'c}$ = $(11110011)_{2'c}$

$(+13)_{10}$ + $(-13)_{10}$ = $(00001101)_{SM}$ + $(11110011)_{2'c}$ = $(11110000)_{2'c}$ = $(00000000)_{SM}$ (1pts)

....

$(+23)_{10}$ = $(00010111)_{SM}$, $(-46)_{10}$ = $(10101110)_{SM}$ = $(11010001)_{1'c}$ = $(11010010)_{2'c}$

$(+23)_{10}$ + $(-46)_{10}$ = $(00010111)_{SM}$ + $(11010010)_{2'c}$ = $(11101001)_{2'c}$ = $(11101000)_{1'c}$ = $(10010111)_{SM}$ = $(-23)_{SM}$

(1pts)

....

$(+127)_{10}$ = $(01111111)_{SM}$, $(+2)_{10}$ = $(00000010)_{SM}$

$(+127)_{10}$ +, $(+2)_{10}$ = $(01111111)_{SM}$ + $(00000010)_{SM}$ = = $(100000001)_{SM}$ with overflow. Invalid result (1pts)

....

Exercise 5 : (05 points)

Let X, Y, Z be input variables, and let F be defined as:

$$\mathbf{F(A, B, C)} = \begin{cases} \mathbf{1}\ \textit{if the majority of variables is } 1 \\ \mathbf{0}\ \textbf{\textit{sinon}} \end{cases}$$

1. Construct the truth table for **F(A, B, C)** .(1pts)

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

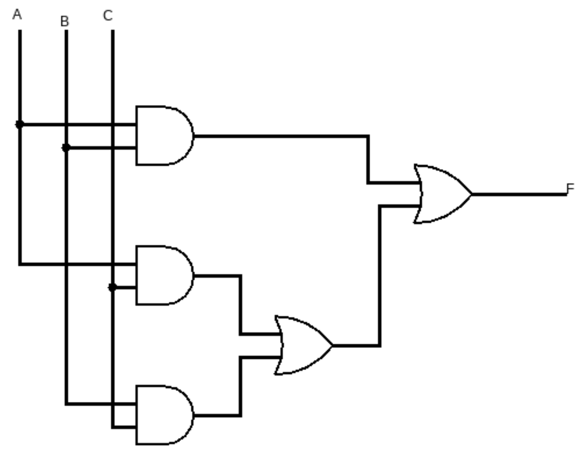
2. Simplify the function F using a Karnaugh map.

C / AB	0	1
00		
01		1
11	1	1
10		1

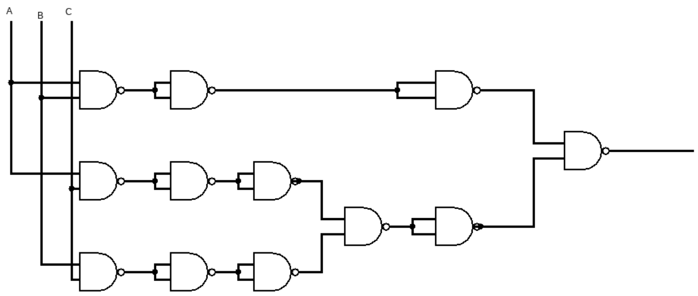
(0.5pts)

F = AB + AC + BC (0.5pts)

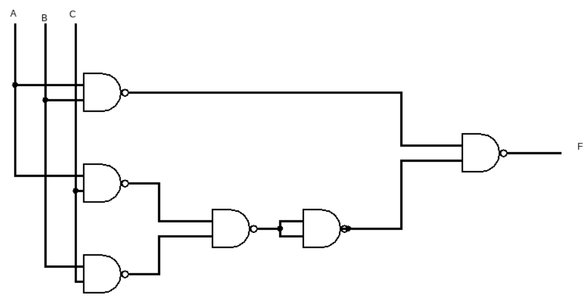
3. Draw the logical circuit implementing F (1pts)



4. Draw the logical circuit implementing F using NAND gates



(1pts)



(1pts)